

PARALLEL DATA INTERFACE

BACKGROUND OF THE INVENTION

The present invention relates to an interface or other apparatus receiving parallel transmitted data streams.

There are generally two well recognised ways in which data is transmitted. In serial data transmission the data is transmitted sequentially via a single transmission channel. In parallel transmission a plurality of associated channels are provided and data is transmitted simultaneously via the plurality of channels.

In any data transmission system, the data is generally transmitted in a fixed relationship to a clock signal. That is, a clock signal defines fixed time slots and one data bit is transmitted in each time slot. Upon reception of the transmitted signal the relationship of the received signal to the data time slots must be established to enable recovery of the transmitted data. Because of variations introduced by the transmission medium it is not possible simply to run a clock having an appropriate frequency at the receiver without ensuring that it is properly synchronised with the incoming data.

In serial transmission systems a suitably synchronised clock at the receive apparatus may be generated from the received data itself, or the data sequence may be used to synchronise a locally generated clock to enable data recovery. Using such arrangements, high data transmission rates have been achieved using serial data transmission technique.

Typically, one channel in a parallel system may be used to transmit a clock signal which can be used for the data recovery at the receiver, and the skew also affects the timing relationship between the clock channel and the data channels.

However, as bandwidth requirements in data transmission systems increase there is demand for the ability to transmit parallel data at data rates in each channel approaching those previously used for serial transmission. At such data rates the problems caused by

skew in the parallel transmission channel have a significant effect in the ability to recover received data.

One approach would be to regenerate a separate data recovery clock for each of the parallel channels. This is however impractical for a large number of parallel data channels, and also does not deal with the lack of synchronisation between the data channels.

SUMMARY OF THE INVENTION

The present invention provides apparatus for receiving parallel transmitted data in a plurality of channels comprising means to generate a clock signal on the basis of the received data and means associated with each of said channels to synchronise data received on the associated channel with the generated clock.

In this arrangement a single clock signal is generated which is used for all the data channels. This means that the apparatus is easily scaleable to receive data from large numbers of parallel channels.

In synchronising all the data channels with a single clock the apparatus also removes the skew between the data channels. Thus the apparatus can simply present as-received but re-aligned data signals for subsequent processing. Alternatively the apparatus can perform the data recovery at the same time as re-aligning the channels.

The clock signal may be generated on the basis of a single received channel. That channel may be a channel designated for the transmission of a clock signal from the transmitter. Alternatively, that channel may be

SECRET

one of the data channels in which it is expected that there will be a significant number of data transitions.

It may also be possible to generate the clock signal on the basis of a plurality of the parallel channels.

The synchronising of each data channel with the clock is preferably done by applying a variable delay to each of the data channels. Also, the generated clock signal is preferably delayed by half the maximum delay available to each data channel so that the data channels can be effectively advanced or retarded in relation to the clock.

BRIEF DESCRIPTION OF THE DRAWINGS

The problems overcome by the invention together with other features and advantages will be more fully explained in the following description of a preferred embodiment, given by way of non-limiting example, and with reference to the accompanying drawings, in which:

Figure 1 shows ideal clock and data signals;

Figure 2 shows clock and data signals with skew;

Figure 3 shows an outline of high-speed parallel interface;

Figure 4 illustrates example phase detector with idealised signal waveforms;

Figure 5 shows a phase detector characteristic;

Figure 6 shows a phase detector characteristic with data delay adjustment range for ideally aligned data;

Figure 7 shows a phase detector characteristic with data delay adjustment range for misaligned data;

Figure 8 shows a phase detector characteristic with high skew and large Td;

Figure 9 shows a phase detector characteristic with high skew and large Td with delay Awrap around@;

Figure 10 illustrates a variable data delay line based on interpolator; and

Figure 11 illustrates an extended data phase interpolator delay line for improved linearity/range.

DETAILED DESCRIPTION

Figure 1 illustrates data signal timing in a typical data transmission system. In particular Figure 1 shows a clock signal 10, known as a half-rate clock, and data slots are defined between clock transitions. This is shown by the representative data stream 12 with sequential data slots 14. In the preferred embodiment it will be assumed that a half-rate clock is transmitted in one of the parallel channels. For data recovery it is usual to re-generate a full-rate clock having a frequency twice that of the half-rate clock which therefore has transitions in the centres of each data slot 14 as well as at the boundaries.

Figure 2 is a diagram similar to Figure 1 but illustrating the effect of skew in the transmission channel. As compared to half-rate clock 10 it can be seen that the boundaries between the data slots in data stream 22 can drift from synchronisation with the clock transitions as a result of variations in the transmission times in the various channels.

More precisely, skew is specified by a single time value representing the maximum alignment error

between any two signals in the parallel transmitted signals. This is defined as T_s and, at worst therefore, any data bit could be shifted early or late with respect to the clock by up to T_s . This receiver needs to be designed to handle such misalignment.

An outline of the preferred parallel interface receiver system is shown in Figure 3. This comprises a Clock Recovery circuit 30 and a set of Data De-skew circuits 40, one for each bit in the parallel bus. The basic principle of the system is to generate a recovered clock from the Clock input and to distribute this to each of the de-skewing circuits where each of the incoming data signals is shifted into alignment with the clock using a variable delay line.

The operation of the De-skew circuits 40 will be described in more detail below, but it may be noted that each such circuit comprises a variable delay 42 which is arranged to apply a variable delay between 0 and T_d to the received data. The delay 42 is controlled by a delay line control means 44 which operates on the basis of a comparison between the delayed data and the clock signal effect by phase detector 46.

A delay line 32 is also used in the clock recovery circuit 30, where it is set to give a delay exactly in the middle of its range: ie the delay line in the clock recovery block is set to $2T_d$. This allows the data to be shifted with respect to the clock by $\pm 2T_d$ in the data de-skewing blocks 40.

The clock recovery system shown is based on a phase interpolation technique wherein an output clock phase is generated from a pair of quadrature reference clocks 35 by summing these with different weightings in a

059691-03001
SECRET

phase interpolator 34. In Figure 3, the reference clocks (and hence the aligned data clock) will nominally be at the full data rate. However, it is possible to adapt the system to operate on a half-rate clock. Control of the phase interpolator 34 is performed using a phase detector 38 to compare the alignment of the recovered clock 50 with the delayed half-rate clock. This then produces control signals which are used to adjust the phase interpolator weightings. The phase interpolator control 36 is generally carried out using digital techniques, although the analogue method described in patent application 0004298.6 may also be used.

The recovered clock 50 is distributed to each of the data channels. In practice, care needs to be taken to ensure that this clock distribution does not itself exhibit skew. The data de-skewing circuits 40 then use phase detectors 46 which may be identical to that in the clock recovery block 30 to control the variable delay lines 42 so as to shift the data into alignment with the recovered clock 50.

The delay lines allow the data to be shifted in position with respect to the clock by $\pm 2T_d$, therefore in order to ensure that the skew can be cancelled out at each input it must be ensured that $2T_d > T_s$.

The precise implementation of the phase detector 38,46 is not a part of this invention. However, in general this will simply provide an indication to either increase the delay (via the "Up" control signal) or decrease the delay (via the "Down" control signal) if the data is early or late respectively. A simple example of a possible phase detector circuit 46 is shown in Figure 4A. This circuit simply samples the received data on the

[illegible][illegible][illegible]

	1987	1988	1989	1990	1991	1992	1993	1994	1995	1996	1997	1998	1999	2000	2001	2002	2003	2004	2005	2006	2007	2008	2009	2010	2011	2012	2013	2014	2015	2016	2017	2018	2019	2020	2021	2022	2023	2024	2025	2026	2027	2028	2029	2030	2031	2032	2033	2034	2035	2036	2037	2038	2039	2040	2041	2042	2043	2044	2045	2046	2047	2048	2049	2050	2051	2052	2053	2054	2055	2056	2057	2058	2059	2060	2061	2062	2063	2064	2065	2066	2067	2068	2069	2070	2071	2072	2073	2074	2075	2076	2077	2078	2079	2080	2081	2082	2083	2084	2085	2086	2087	2088	2089	2090	2091	2092	2093	2094	2095	2096	2097	2098	2099	2100	2101	2102	2103	2104	2105	2106	2107	2108	2109	2110	2111	2112	2113	2114	2115	2116	2117	2118	2119	2120	2121	2122	2123	2124	2125	2126	2127	2128	2129	2130	2131	2132	2133	2134	2135	2136	2137	2138	2139	2140	2141	2142	2143	2144	2145	2146	2147	2148	2149	2150	2151	2152	2153	2154	2155	2156	2157	2158	2159	2160	2161	2162	2163	2164	2165	2166	2167	2168	2169	2170	2171	2172	2173	2174	2175	2176	2177	2178	2179	2180	2181	2182	2183	2184	2185	2186	2187	2188	2189	2190	2191	2192	2193	2194	2195	2196	2197	2198	2199	2200	2201	2202	2203	2204	2205	2206	2207	2208	2209	2210	2211	2212	2213	2214	2215	2216	2217	2218	2219	2220	2221	2222	2223	2224	2225	2226	2227	2228	2229	2230	2231	2232	2233	2234	2235	2236	2237	2238	2239	2240	2241	2242	2243	2244	2245	2246	2247	2248	2249	2250	2251	2252	2253	2254	2255	2256	2257	2258	2259	2260	2261	2262	2263	2264	2265	2266	2267	2268	2269	2270	2271	2272	2273	2274	2275	2276	2277	2278	2279	2280	2281	2282	2283	2284	2285	2286	2287	2288	2289	2290	2291	2292	2293	2294	2295	2296	2297	2298	2299	2300	2301	2302	2303	2304	2305	2306	2307	2308	2309	2310	2311	2312	2313	2314	2315	2316	2317	2318	2319	2320	2321	2322	2323	2324	2325	2326	2327	2328	2329	2330	2331	2332	2333	2334	2335	2336	2337	2338	2339	2340	2341	2342	2343	2344	2345	2346	2347	2348	2349	2350	2351	2352	2353	2354	2355	2356	2357	2358	2359	2360	2361	2362	2363	2364	2365	2366	2367	2368	2369	2370	2371	2372	2373	2374	2375	2376	2377	2378	2379	2380	2381	2382	2383	2384	2385	2386	2387	2388	2389	2390	2391	2392	2393	2394	2395	2396	2397	2398	2399	2400	2401	2402	2403	2404	2405	2406	2407	2408	2409	2410	2411	2412	2413	2414	2415	2416	2417	2418	2419	2420	2421	2422	2423	2424	2425	2426	2427	2428	2429	2430	2431	2432	2433	2434	2435	2436	2437	2438	2439	2440</
--	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	--------

increased data delay adjustment range. Under these conditions, it is possible to adjust the phase of the data such that it overlaps into the adjacent bit period. If the system were to get into this state, the phase detector 46 would indicate the wrong direction to centre the data (e.g in Figure 8, the phase detector would try to increase the delay rather than reduce it) and would potentially lock up at the end stop of the delay line range. It can be seen that the condition for this to occur is that $T_s + 2T_d > 2UI$.

The range for Td to meet these requirements is therefore as follows:-

$$T_s < 2T_d < (2U_I - T_s)$$

These constraints could prove a serious limit to the practicality of this system in reality, since T_d will be subject to variation due to manufacturing tolerances, whilst any increase in T_s results in a decrease in the tolerable range of T_d for both its minimum and maximum values. For instance, if $T_s = 3UI$, T_d has zero margin for error.

In order to alleviate these constraints, it is desirable to avert the potential lock-up condition. In fact it is possible to do this by allowing the delay line control to Awrap around@ from its maximum value to its minimum value and vice versa. If this is implemented, no potential lock-up will occur unless the skew and data delay are sufficient for it to lock onto the centre of the adjacent data bit as shown in Figure 9. This will only occur if $T_s + 2T_d > 2UI$. Thus our restrictions for T_d are now as follows:

$$T_s < 2T_d < (U_i - T_s)$$

